In the Specification:

Page 8, after line 2, after the insert as submitted with the Supplemental Response to Office Action filed August 23, 2006, please insert the following new paragraph:

--Figure 8 shows a sample implementation for a state storage flag.--

Page 12, after line 15, after the insertion of a replacement section of text as submitted with the Supplemental Response to Office Action filed August 23, 2006, please add the following new section:

--Discussion from Incorporated Nonprovisional Application No. 10/373,628 Filed 02/24/2003

--This invention is a methodology for enhancing RFID performance when identifying or writing to two or more tags. Both methodologies become crucial to performance when a substantial number of tags, for example ten or more, are being identified or written. The methodology is embodied by two commands that will specifically select RFID tags based on certain selection criteria. The criteria for selection can be set based on user requirements and the user can do the following operations
1)selection of any combination of a subset of
available flags
2)selection based on matching flag condition
3)selection based on non-matching flag condition

--The flags that are currently available are state_storage flag and write_ok flag. The state storage flag indicates whether or not the tag was in a specific state(DATA_EXCHANGE_STATE) prior to losing power and the write_ok flag indicates if the last write operation on the EEPROM was done with adequate power supply(i.e. a good write was done into the EEPROM memory matrix).

--The two commands that perform the selection of tags are
1)Group select flags- this will move tags from the
READY state to the ID state
2)Group unselect flags- this will move tags from the
ID to the READY state.
Both command will do their respective operations only
if the flags on a tag match the selection criteria.

--Detailed description of the group select flags
commands is as follows-

--The various fields for group_select_flags for selecting on the write_ok flag and the state_storage flag are as follows:-

<preamble> <command> <bit_mask> <data> <crc>

--Both the bit_mask and the data fields are one byte fields. The bit_mask will enable selection on flags and once a bit flag is enabled, the value of the data field will enable selection on flag high or low. For example, if the last two bits of the bit_mask and the data field are used for state_storage and write_ok(LSB) in that order, then a few scenarios are as follows:- (only last two bits for each field are shown)

BM Data

11 11 - will select all tags with state_storage set and write_ok set

11 01 - will select all tags with state_storage not set and write_ok set

01 11 - will select all tags with state_storage set

--The original description of the invention follows:

--REVIEW OF CHIP STATE STORAGE CAPABILITY

--OBJECTIVE

--Tags losing power while in the middle of the Multi-Item protocol add enormous latency to the protocol efficiency. Tags can lose power while in the ID state or the DATA_EXCHANGE state. Intuitively tags loosing power in the DATA_EXCHANGE add longer times to the protocol because they have to go through the ID state (without loosing power) and finally end in the DATA_EXCHANGE state; whereas tags loosing power in the ID state add lesser time overhead relatively. Thus if a technique that could prevent tags that were in the DATA_EXCHANGE state (and lost power)from coming back into the protocol to be identified, this would save considerable time. It must be noted that it is difficult to delineate the exact amount of overhead that the two situations (tags loosing power in the ID state, tags loosing power in the DATA_EXCHANGE state) contribute by themselves.

--REQUIREMENTS FROM A SYTEM STANDPOINT

1) To identify all the tags within the range of a reader (regardless of whether they were identified before or not)

2) To identify only tags that were identified (brought to DATA_EXCHANGE) but subsequently lost power

3) To avoid inclusion of tags that were identified once back into the protocol loop once again

--SOLUTION

--The above three requirements can be met with the following solution.

The tag have the capability of storing a voltage ($V_{STORAGE}$) on a high impedance node (e.g. high impedance node 847, Fig. 8) that is charged high when the tag goes to DATA_EXCHANGE state- the voltage would also be discharged when an INITIALISE command or an appropriate GROUP_SELECT command is issued from the reader.

--REVIEW OF CHIP STATE MACHINE ARCHITECTURE WITH THE ABOVE MODIFICATION

--The Gamma ASIC has three major states- READY, ID and DATA_EXCHANGE. With the above modification, the following table indicates the storage value vs. tag state

TAG STATE	$V_{STORAGE}$
READY	Can be high or low
ID	Low
DATA_EXCHANGE	High

$V_{STORAGE}$ is high in the READY state if the tag was previously identified and lost power and went back into the ready state.

--ADDITIONAL COMMANDS REQUIRED

--Additional commands have to be added to the capability of the reader to enable selection/non-selection of tags with $V_{STORAGE}$ high/low

GROUP_SELECT_EQ_SS- This command will select only tags that have $V_{STORAGE}$ high.

GROUP_SELECT_EQ_NSS-This command will select only tags that do not have $V_{STORAGE}$ high(tags that were not identified, or tags which had $V_{STORAGE}$ high but the voltage discharged off, or tags which had the high $V_{STORAGE}$ reset to low with an INTIALISE command)

--Note that a GROUP_SELECT_EQ command will select all the tags; and all other commands will not differentiate between a tag that has $V_{STORAGE}$ high vs. $V_{STORAGE}$ low.--

(The foregoing subject matter was incorporated herein by reference in its entirety as APPENDIX A in the present case as filed.)

--Incorporation By Reference

--U.S. Patents 5,550,547, 5,850,181 and 5,673,037 are hereby incorporated by reference as providing background information to assist in understanding the foregoing disclosure.

--U.S. Patent 6,404,325, which is incorporated herein by reference, shows circuitry in Figs. 4 and 5 which may be utilized to serve as a state storage flag, e.g. as shown in Fig. 8, to indicate whether the tag was or was not in a specific state, e.g. DATA_EXCHANGE state. For example, a capacitor such as C(Aux) in Fig. 4 may be charged only when the tag goes to the DATA_EXCHANGE state prior to loosing power. The check circuit and flip-flop 41 of Figure 5 may provide the state storage flag output at line 43 from the high impedance node that is charged high when the tag goes to DATA_EXCHANGE state. The voltage on the high impedance node would be discharged when an INITIALISE command or an appropriate GROUP_SELECT command is issued from the reader.--

Page 12, after the foregoing insertion, please insert the following new section:

--Description of Fig. 8

--U.S. Patent 6,404,325, which is incorporated herein by reference, shows circuitry in Figs. 4 and 5 which may be utilized to serve as a state storage flag, e.g. as shown in Fig. 8, to indicate whether the tag was or was not in a specific state, e.g. DATA_EXCHANGE state. For example, a capacitor such as 8-CAux in Fig. 8 may be charged only when the tag goes to the DATA_EXCHANGE state prior to losing power. A check circuit 842 and flip-flop 841 of Fig. 8 may provide the state storage flag output at line 843 from the high impedance node 847 that is charged high when the tag goes to DATA_EXCHANGE state. The voltage on the high impedance node 847 would be discharged when an INITIALISE command or an appropriate GROUP_SELECT command is issued from the reader.--